

# 스마트 교통 안전 서비스

## AI 기반 위험도 예측 시스템

팀 : 황수용, 문운호, 이기표

발표자: 황수용

발표일: 2025년 6월 26일



# 교통안전 문제의 심각성과 과제

## 인명 피해의 심각성

매년 수많은 사망자와 부상자 발생, 특히 보행자, 고령자, 이륜차 운전자가 고위험군에 속함

WHO 보고: 교통사고는 15~29세 청년층 사망 원인 1위

## 막대한 사회적 비용

의료비, 사고처리 비용, 근로손실, 보험료 상승 등 직·간접 비용 발생

한국교통안전공단: 연간 사회적 비용 약 30조 원 추산

## 교통약자 보호 미흡

고령자, 어린이, 장애인 등 교통약자 사고 취약성 높음

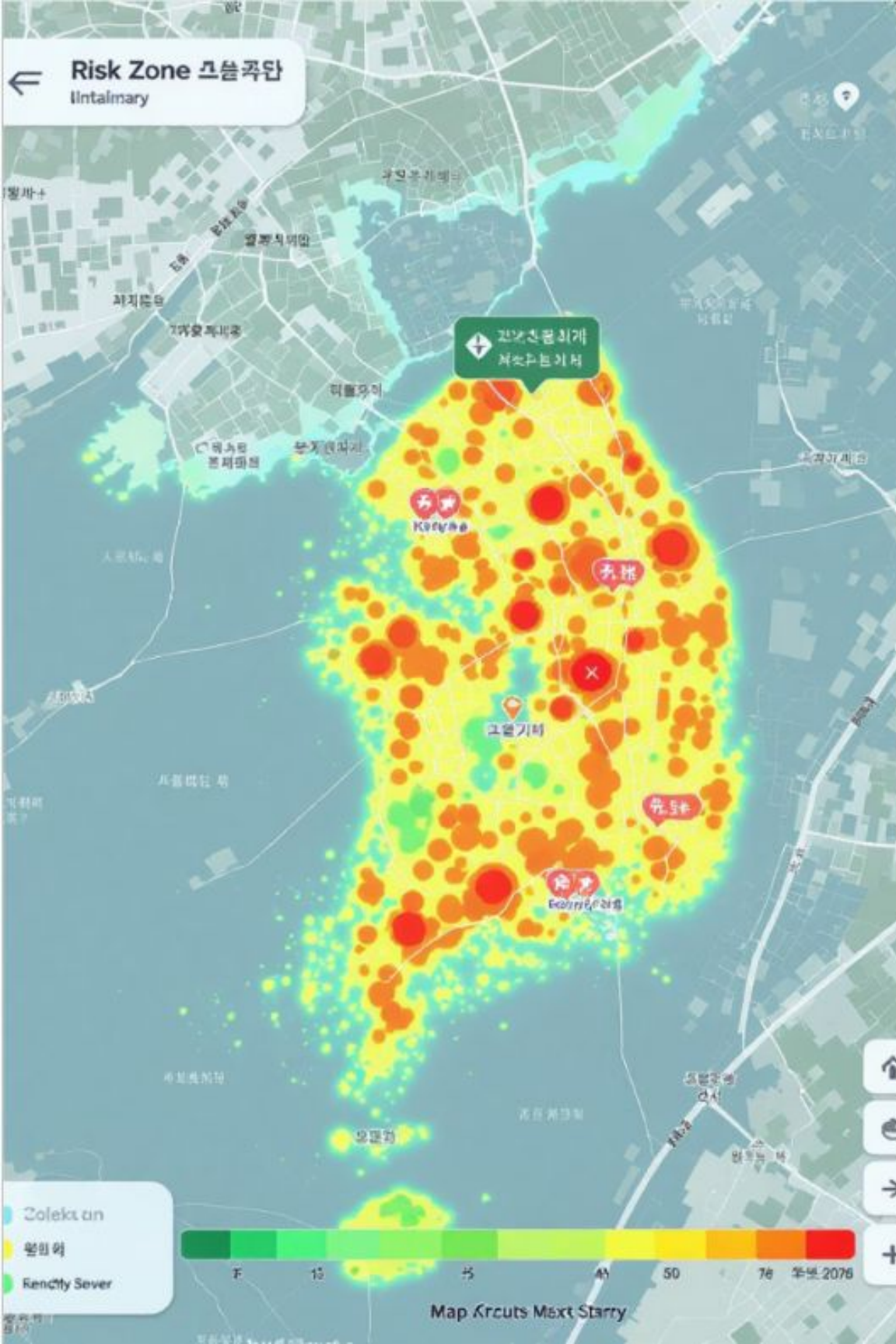
보호 인프라 부족 지역 존재, 스쿨존 사고는 사회적 공분 초래



## 현재 상황

- 보행자 교통사고는 날씨, 시간대, 운전 유형에 따라 크게 변동
- 기존 안전 시스템은 사고 예방보다 사후 처리에 집중
- 사고 발생 패턴에 대한 체계적 분석 부족
- 실시간 위험 정보 제공 체계 미비





# 프로젝트 목표



## 예측 시스템 구축

사고 발생 가능성이 높은 지역과 시점을  
정확하게 예측하는 AI 기반 시스템 개발



## 사전 경고 체계

위험 상황 발생 전 운전자와 보행자에게  
실시간 알림 제공



## 위험 지도 제공

지역별, 시간대별 위험도를 시각화한  
직관적인 지도 서비스 구현



## 서비스 주요 기능



### 지역별 위험 지도

색상 코드로 구분된 직관적인 위험 지도  
제공, 실시간 업데이트



### 실시간 위험도 예측

현재 상황과 과거 데이터를 결합한 정확한  
위험도 산출 및 표시



### 위치 기반 경고 알림

사용자 위치에 따른 맞춤형 경고 메시지 및  
안전 경로 추천



# 스마트 보행 안내 서비스, 제주도가 최적의 시작점

## 왜 제주특별자치도인가?

- 관광객 중심의 보행 친화적 도시 구조
- 협소하고 복잡한 도로 구조로 정확한 경로 안내 수요 높음
- 풍부한 공공데이터와 스마트도시 실증 경험 보유
- 지리적으로 한정된 섬 지역으로 테스트베드 최적화

제주도는 안전한 보행 경로 안내 서비스를 실현할 수 있는 최적의 출발점입니다.



제주도의 독특한 지리적 특성과 관광 인프라는 스마트 보행 서비스의 시범 운영에 이상적인 환경을 제공합니다.

### 1단계: 관광 중심지 시범 운영

성산일출봉, 올레길 등 관광객 밀집 지역 중심으로 서비스 도입

### 2단계: 제주 전역 확대

시내 중심지 및 주요 생활권으로 서비스 영역 확장

### 3단계: 타 지역 모델화

제주 성공 사례를 기반으로 전국 다른 관광지 및 도시로 확산





# 교통사고 통계와 스마트시티 구축 기반

## 교통사고 통계 현황

제주도는 보행자 대상 교통사고 비율이 높은 지역 중 하나입니다.

특히 야간 사고율이 높고, 관광객의 지리 정보 부족으로 인한 사고 위험이 큼니다.

이는 **AI** 기반 경로 추천 시스템의 필요성을 뒷받침합니다.



## 스마트시티 추진 기반

제주특별자치도는 스마트관광도시 1차 시범지역으로 선정된 바 있어

**AI** 기반 도시 안전 서비스와의 궁합이 우수합니다.

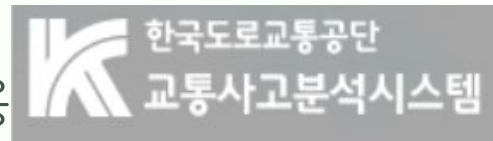
공공 데이터 개방이 활발하여 서비스 구현 및 확장에 필요한 데이터 접근성이 우수합니다.



# 데이터 소개 및 개발 환경

## 활용 데이터

- 도로교통공단 TAAS 사고 데이터(학습용)
- 제주도 교통사고 데이터(예측용)
- 기상청 날씨 예보 OpenAPI(예측용)



## 개발 언어 및 프레임워크



# 데이터 특징 및 분석 전략

## 1. 피쳐 특징

- object 타입의 피쳐 다수로 이루어짐
- int 타입의 피쳐  
(사망자수+중상자수+경상자수+부상신고자수)  
조합으로 심각도 점수 도출  
**데이터의 분포가 심하게 왼쪽으로 치우침**  
→ 대부분 값이 작고, 일부 값만 매우 큼  
→ 로그변환으로 처리
- object 타입의 피쳐를 원핫인코딩 후 모델에 입력

## 2. 모델 선택

- 원핫인코딩과 맞는 트리형 회귀모델 선택

|            |       |          |        |
|------------|-------|----------|--------|
| 구분번호       | 16997 | non-null | int64  |
| 발생년월       | 16997 | non-null | object |
| 주야         | 16997 | non-null | object |
| 시군구        | 16997 | non-null | object |
| 사고내용       | 16997 | non-null | object |
| 사망자수       | 16997 | non-null | int64  |
| 중상자수       | 16997 | non-null | int64  |
| 경상자수       | 16997 | non-null | int64  |
| 부상신고자수     | 16997 | non-null | int64  |
| 사고유형       | 16997 | non-null | object |
| 법규위반       | 16997 | non-null | object |
| 노면상태       | 16997 | non-null | object |
| 기상상태       | 16997 | non-null | object |
| 도로형태       | 16997 | non-null | object |
| 가해운전자 차종   | 16997 | non-null | object |
| 가해운전자 성별   | 16997 | non-null | object |
| 가해운전자 연령대  | 16997 | non-null | object |
| 가해운전자 상해정도 | 16997 | non-null | object |
| 피해운전자 차종   | 16997 | non-null | object |
| 피해운전자 성별   | 16997 | non-null | object |
| 피해운전자 연령대  | 16997 | non-null | object |
| 피해운전자 상해정도 | 16997 | non-null | object |



# 데이터분석 및 예측 전략(1/2)



## 광역시 사고 데이터 피쳐 분석

object 타입의 피쳐가 다수



## 심각도 점수 피쳐생성

생성된 심각도 점수를 예측하고 점수 기반 저위험 / 중위험 / 고위험 구분



## 원-핫 인코딩으로 벡터화

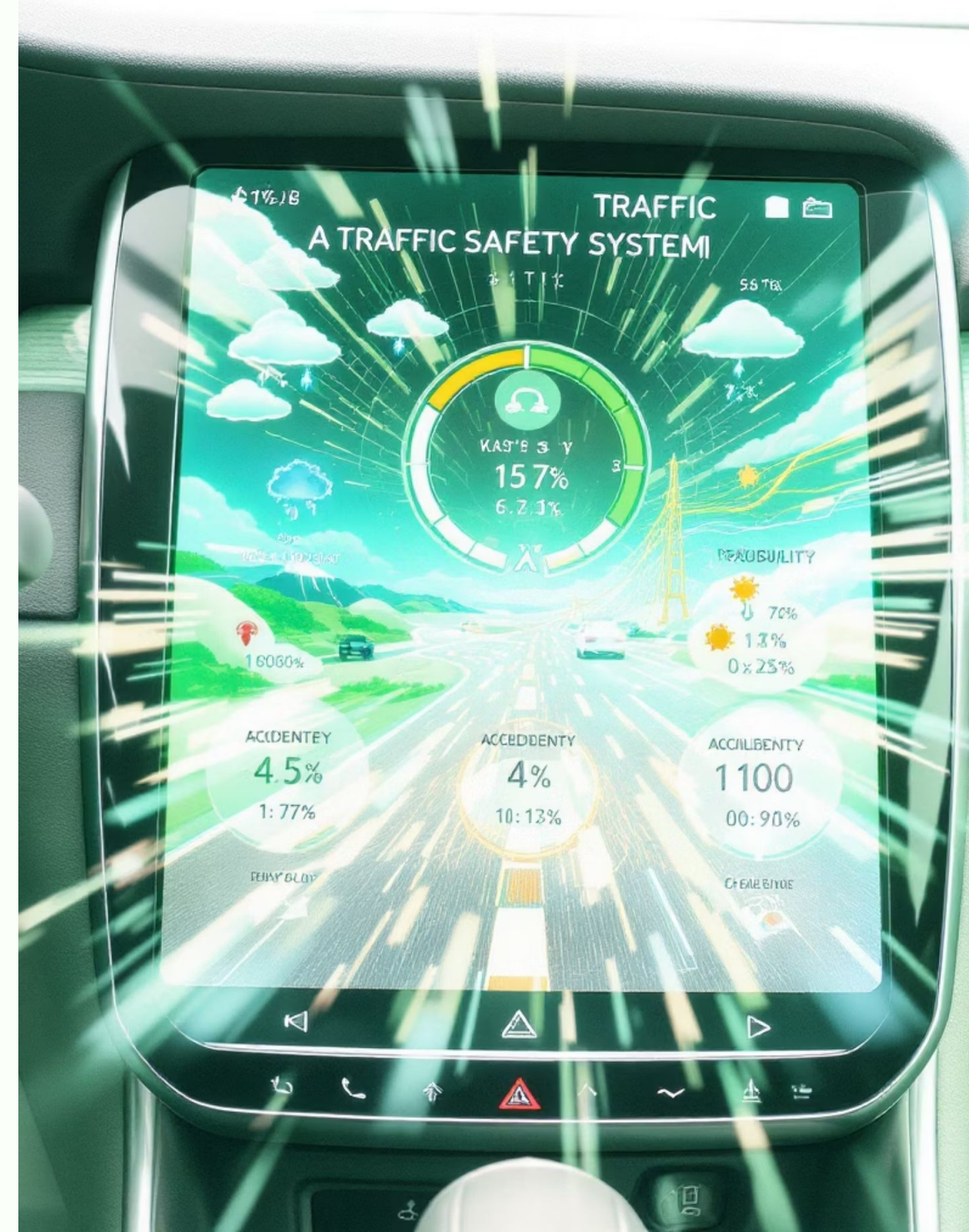
범주형 피쳐에 대해 원-핫 인코딩을 적용하여 벡터화한 뒤, 회귀

예측에는 트리 계열 모델을 활용



## 오버 샘플링

저위험, 고위험 데이터의 부족으로 오버샘플링 시도





# 데이터분석 및 예측 전략(2/2)



가장 성능이 좋은 모델 선택

L회귀 모델 기반 심각도 점수 예측



실시간 데이터 반영

현재 시간(주야간), 월, 날씨, 도로상태 등을 동적으로 반영



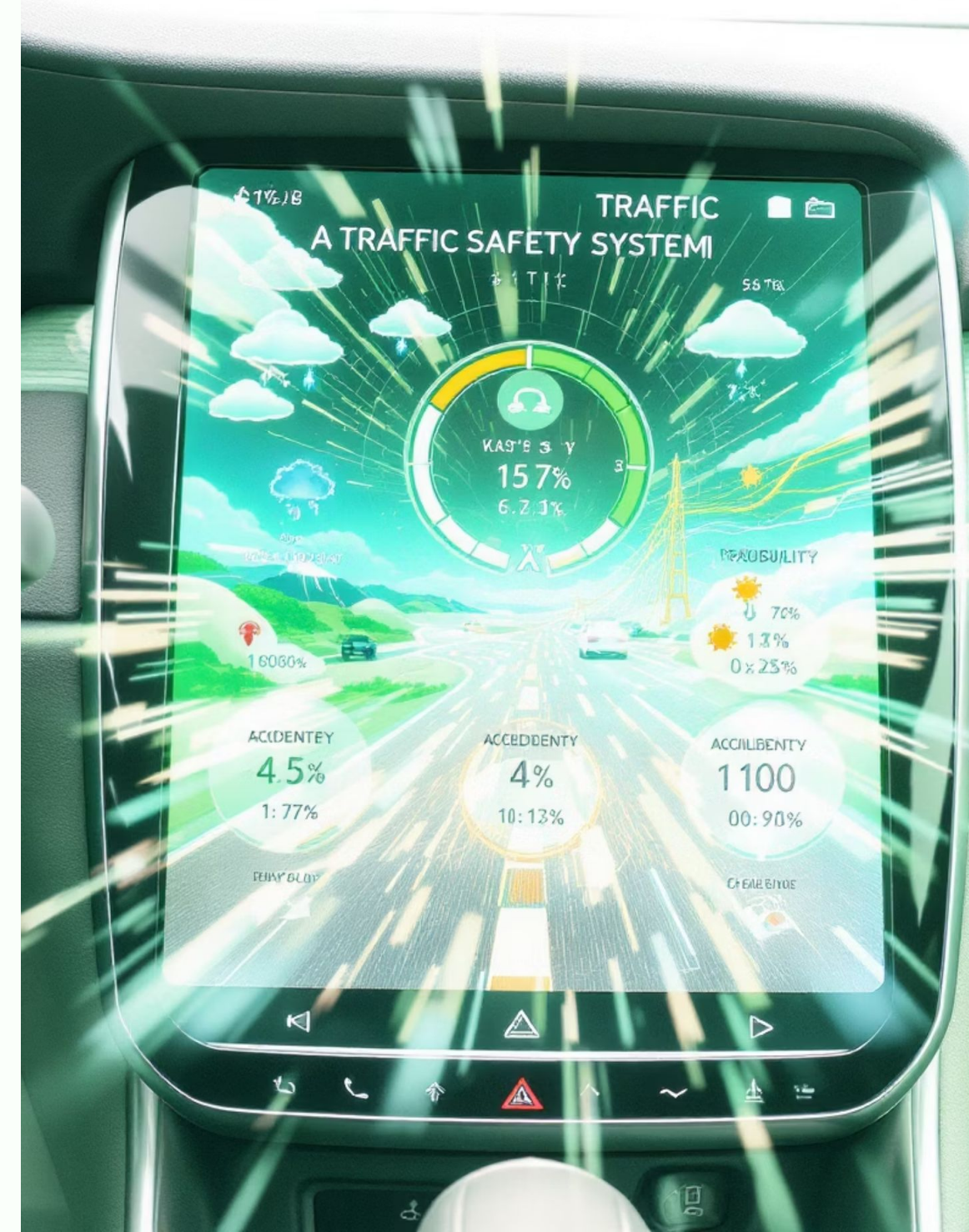
위험도 분류

점수 기반 저위험 / 중위험 / 고위험 구분



시각화 및 저장

위험도 결과를 DB 저장 및 지도 시각화





# 회귀 모델 선택

- 심각도 점수 = 사망자수\*5 + 중상자수\*3 + 경상자수 \*2 + 부상신고자수\*0.5 으로 만듬  
ANSI / KABCO방식(통계적 심각도 분석에 적합)
- 심각도 점수를 타겟피쳐로 점수 예측하는 모델 생성 (RandomForest + GradientBoosting)

RandomForest 모델 평가:

MAE: 0.3360

RMSE: 0.5449

R2: -0.1362

GradientBoosting 모델 평가:

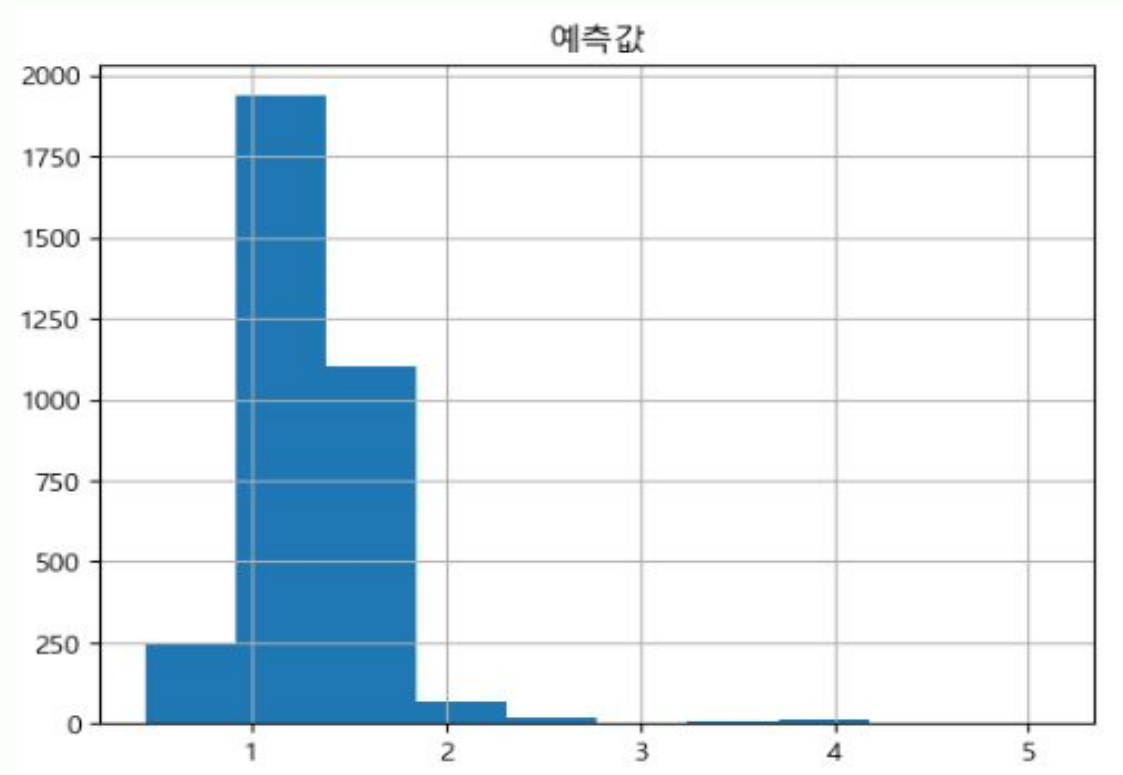
MAE: 0.3112

RMSE: 0.4830

R2: 0.1071

# 모델링 문제점 해결

- 모델의 설명력을 높이기 위해 기존 피쳐 외에 추가적인 유의미한 피쳐들을 생성하여 학습에 활용하였고,  
그 결과  $R^2$  점수가 향상되었습니다.
- 심각도 점수 = 사망자수\*5 + 중상자수\*3 + 경상자수 \*2 +  
부상신고자수\*0.5
  - 대부분 값이 작고, 일부 값만 매우 큼
  - 점수를 낮춰서 분포의 차이를 줄임
  - 로그 변환으로 처리





# 회귀 모델 다시 선택

- RandomForest + GradientBoosting + XGBoost + LightGBM + CatBoost 모델 선택으로 심각도 점수를 예측
- 모든 모델의  $R^2$ 가 향상 됨  
→ 모델이 실제 값을 더 잘 예측(설명력이 올라감)
- $R^2$  값이 준수한 LightGBM과 CatBoost를 비교

RandomForest 모델 평가:

MAE: 0.1028

RMSE: 0.4347

R2: 0.2768

GradientBoosting 모델 평가:

MAE: 0.0901

RMSE: 0.3540

R2: 0.5205

XGBoost 모델 평가:

MAE: 0.1046

RMSE: 0.4834

R2: 0.1057

[LightGBM] [Warning] Found whitespace in feature\_names, replace with underlines

[LightGBM] [Info] Auto-choosing row-wise multi-threading, the overhead of testing was 0.000483 seconds.

You can set `force\_row\_wise=true` to remove the overhead.

And if memory is not enough, you can set `force\_col\_wise=true`.

[LightGBM] [Info] Total Bins 157

[LightGBM] [Info] Number of data points in the train set: 13597, number of used features: 73

[LightGBM] [Info] Start training from score 1.218283

LightGBM 모델 평가:

MAE: 0.0907

RMSE: 0.3245

R2: 0.5970

CatBoost 모델 평가:

MAE: 0.0904

RMSE: 0.3674

R2: 0.4835

# LightGB Model 스코어 분석

| 지표             | 값      | 해석                                     |
|----------------|--------|--|
| MAE            | 0.0907 | 평균적으로 약 0.0907의 오차 발생                  |
| RMSE           | 0.3245 | 예측값의 평균 오차가 0.32 정도로, 큰 오차일수록 더 크게 반영됨 |
| R <sup>2</sup> | 0.5970 | 모델이 약 60% 정도의 변동성을 설명함 (중간 이상 성능)      |

- **MAE가 낮다** → 예측값이 전반적으로 실제값과 **크게 벗어나지 않음** → 정밀한 예측이 이루어졌다는 신호
- **RMSE > MAE** → 예측값 중 큰 오차가 일부 존재함 → RMSE가 MAE보다 크면, 일부 극단적 오차가 있다는 것
- 현재 모델은 양호한 수준이지만, 추가적인 성능 향상을 위해 피처 엔지니어링, 하이퍼파라미터 튜닝 등을 고려할 수 있습니다.



# CatBoost Model 스코어 분석

| 지표                             | 값      | 해석                             |
|--------------------------------|--------|--------------------------------|
| MAE (Mean Absolute Error)      | 0.0904 | 예측값은 실제값과 평균적으로 0.0904 정도 차이   |
| RMSE (Root Mean Squared Error) | 0.3674 | 예측 오차의 표준적인 크기는 약 0.3674       |
| R <sup>2</sup> (결정계수)          | 0.4835 | 모델이 전체 데이터의 약 48.35%의 변동성을 설명함 |

- **MAE가 낮다** → 예측값이 전반적으로 실제값과 크게 벗어나지 않음 → 정밀한 예측이 이루어졌다는 신호
- **RMSE > MAE** → 예측값 중 큰 오차가 일부 존재함 → RMSE가 MAE보다 크면, 일부 극단적 오차가 있다는 것
- **R<sup>2</sup> = 0.4835** → 설명력이 **50% 미만**으로, 모델이 전체 데이터를 절반 정도만 설명하고 있음  
→ 개선 여지 있음 (예: 더 많은 피처, 데이터 정제, 모델 튜닝 등)

# LightGBM vs CatBoost

| 항목         | LightGBM (LGBM)                      | CatBoost                       |
|------------|--------------------------------------|--------------------------------|
| 출시 주체      | Microsoft                            | Yandex (러시아 검색 엔진 기업)          |
| 데이터 처리 속도  | 매우 빠름 (대용량 데이터에 최적)                  | 빠름 (LGBM보다는 느림)                |
| 범주형 처리     | 수동 변환 필요 (One-hot, Label Encoding 등) | 자동 처리 (CatBoost의 가장 큰 장점 중 하나) |
| 결정 트리 구조   | Leaf-wise (depth 제한) → 더 빠르고 깊은 트리   | Symmetric tree → 구조 균형 유지      |
| 과적합 가능성    | Leaf-wise 구조로 인해 과적합 위험 ↑            | 구조적으로 과적합에 강함                  |
| 하이퍼파라미터 튜닝 | 상대적으로 어려움                            | 비교적 쉬움 (기본값이 잘 설정되어 있음)        |
| 성능         | 대용량 데이터에서 빠르고 강력한 성능                 | 범주형 변수 및 소규모 데이터에서 강력          |
| GPU 지원     | 지원                                   | 지원 (더 최적화되어 있음)                |

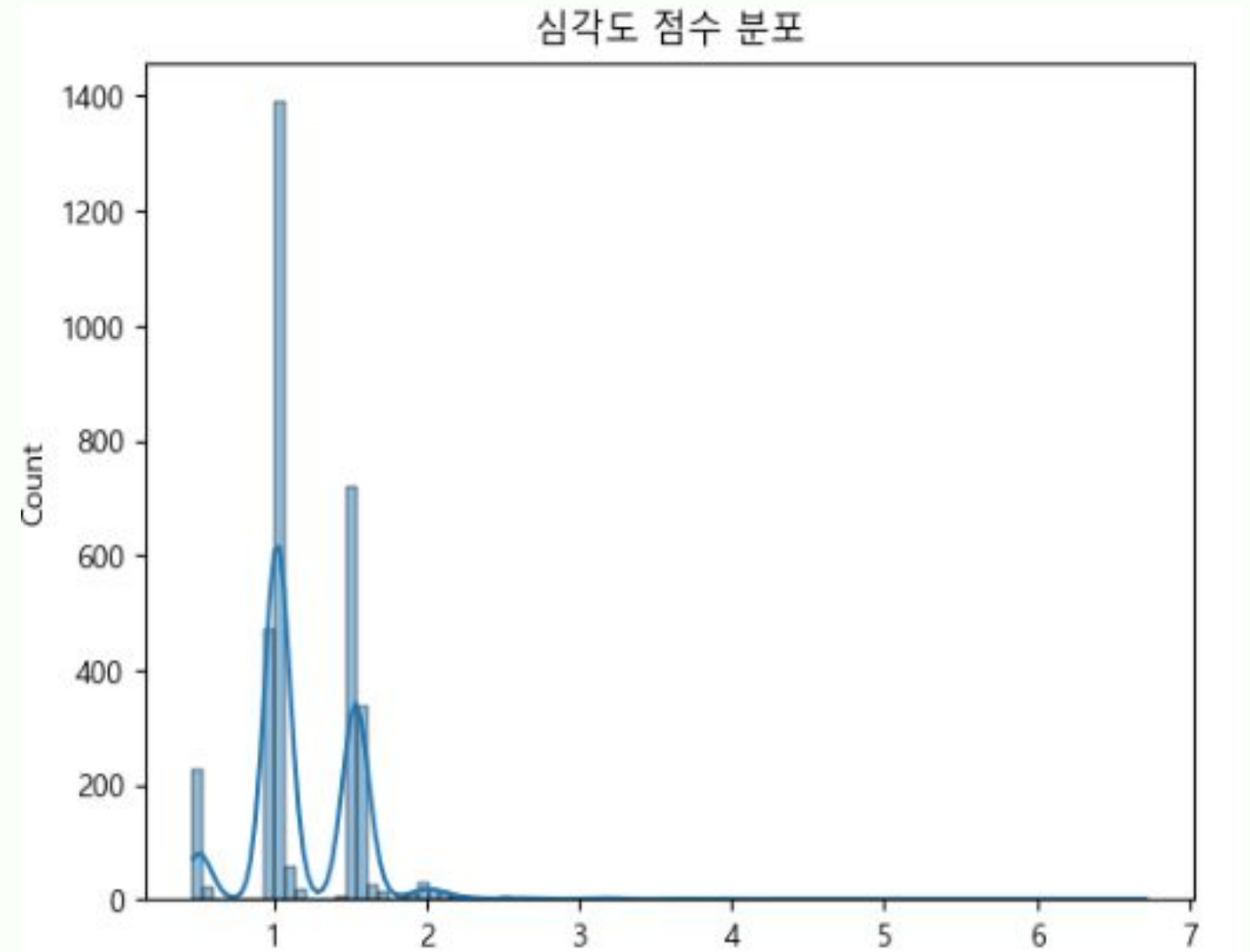
- 성능이 좀더 좋은 LightBGM 모델 선택  
(MAE는 두 모델이 거의 동일 (0.090대)RMSE는 LightGBM이 더 낮아 → 큰 오차를 더 잘 억제  
R²는 LightGBM이 약 60% 설명력, CatBoost는 48%)

| 지표 이름            | 의미 / 수식                                   | 해석 방법                | 특징 및 유의점                  |
|------------------|---|----------------------|---------------------------|
| MAE (평균 절대 오차)   | Mean Absolute Error:<br>$MAE = \text{평균}$ | 실제값과 예측값의 차이의 절대값 평균 | 이상치에 둔감 (평균 오차 직관적 해석 가능) |
| MSE (평균 제곱 오차)   | Mean Squared Error:<br>$MSE = \text{평균}$  | 차이의 제곱 평균            | 이상치에 민감 (오차가 클수록 더 큰 패널티) |
| RMSE (평균 제곱근 오차) | Root Mean Squared Error: $\sqrt{MSE}$     | MSE의 제곱근             | 단위 해석 쉬움, MSE와 동일한 민감도    |
| R² (결정계수)        | R-squared: 설명력 지표 (0~1)                   | 1에 가까울수록 설명력 높음      | 음수도 가능 (예측이 평균보다 못한 경우)   |



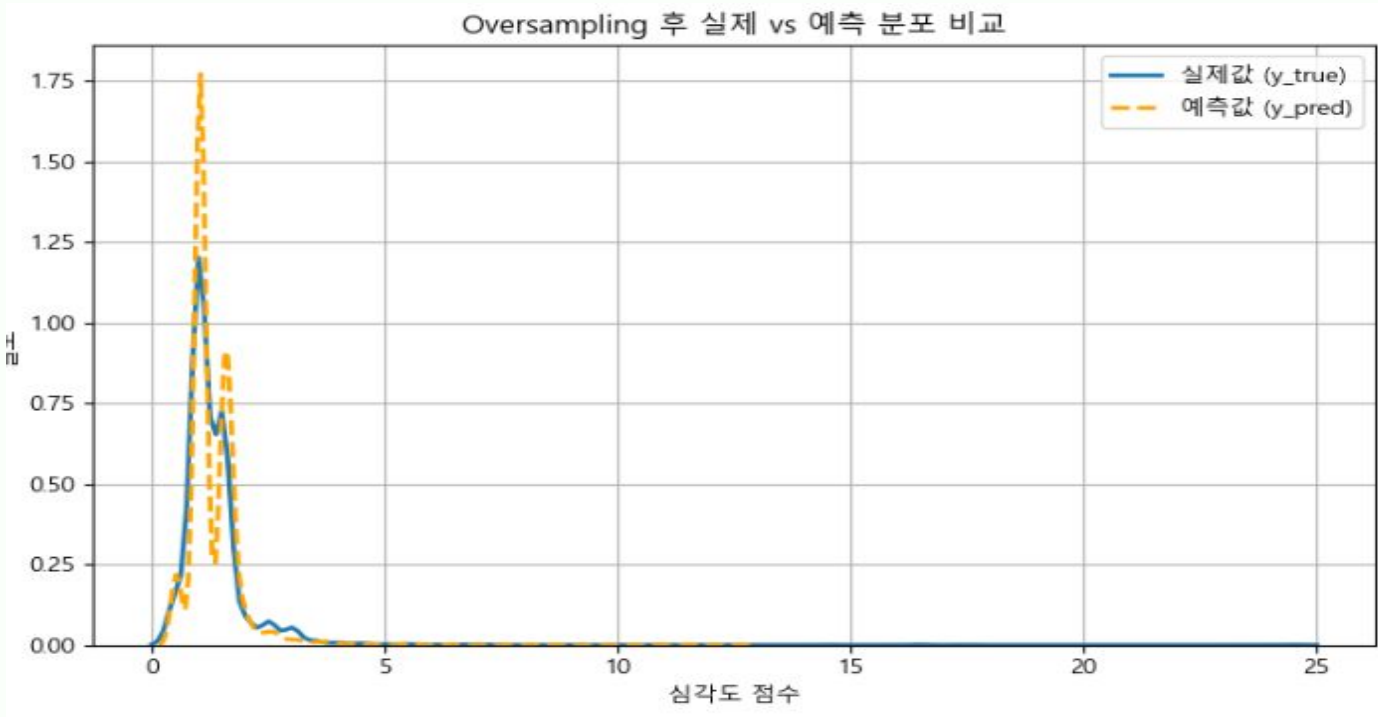
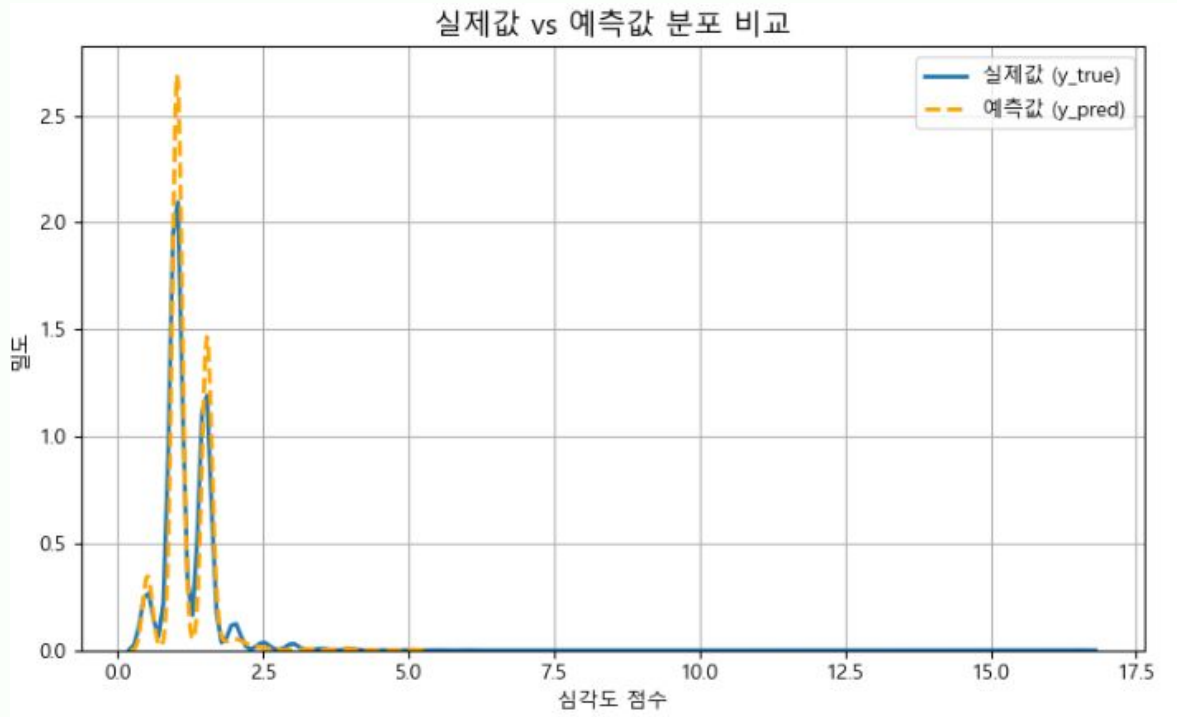
# 예측 분석

- 대부분의 데이터가 심각도 점수 1~2 사이에 몰려 있음
- 우측으로 긴 꼬리 (right-skewed) → 점수 3 이상은 매우 희귀
- 즉, 저심각도 사고 데이터가 압도적으로 많고, 고심각도 사고(중대 사고)는 매우 적음



# 오버샘플링으로 점수를 올려보자

- **Oversampling: 고심각도 사고 5% 이상을 2배 복제**



| 항목        | 오버샘플링 전        | 오버샘플링 후        |
|-----------|----------------|----------------|
| 분포 유사도    | 낮음 (좁은 범위에 치중) | 높음 (고점도 일부 반영) |
| 고심각도 예측   | 거의 없음          | 일부 반영          |
| 분포 왜곡     | 있음 (평균 쏠림)     | 완화됨            |
| 실제 분포 반영력 | 낮음             | 개선             |

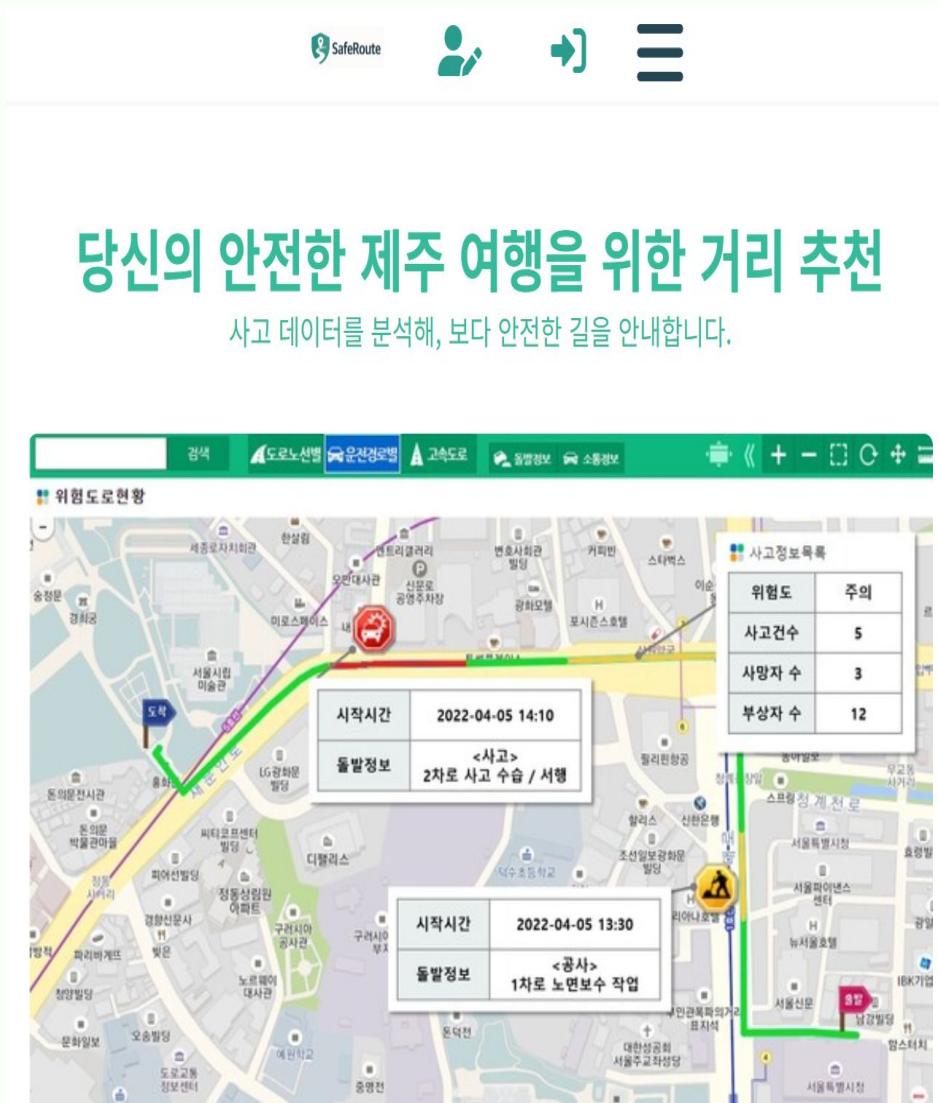
[LightGBM + 고심각도 Oversampling 모델 평가]

MAE : 0.1367

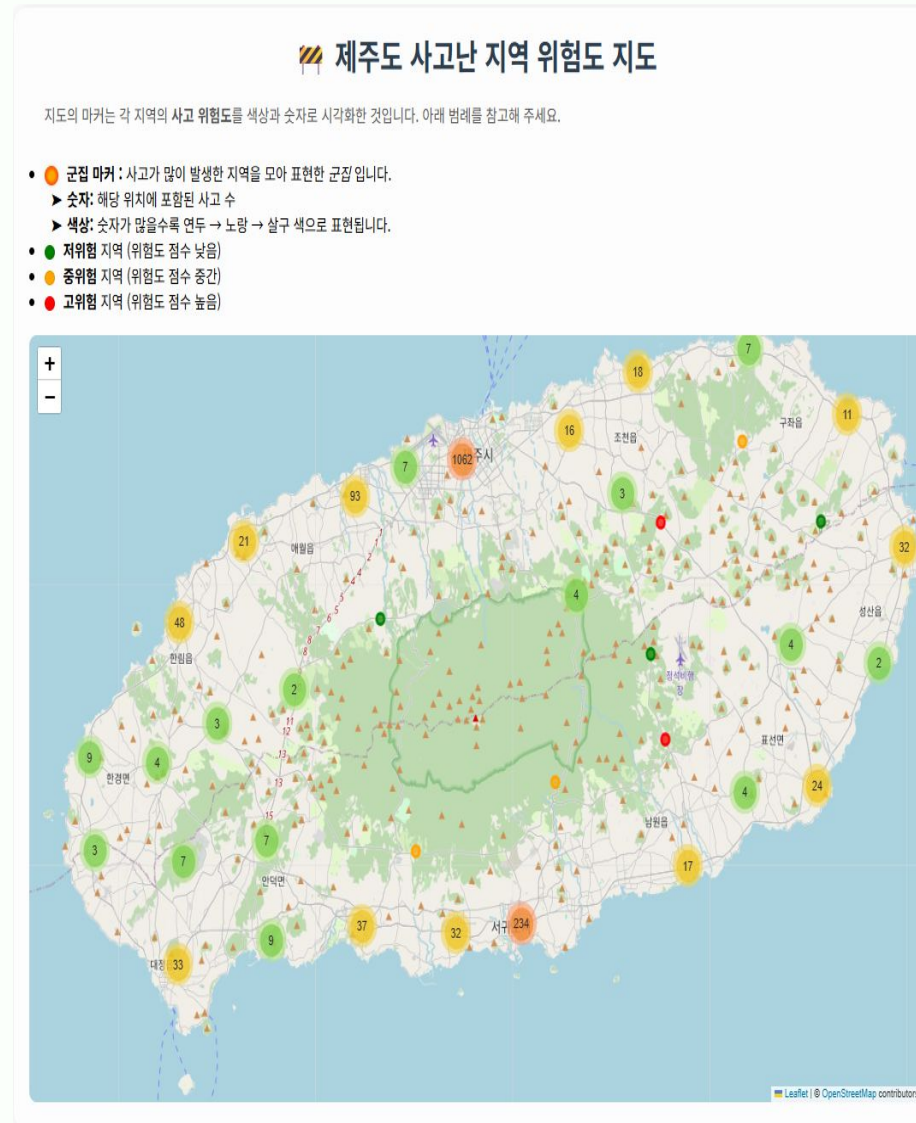
RMSE : 0.4885

R2 : 0.7036

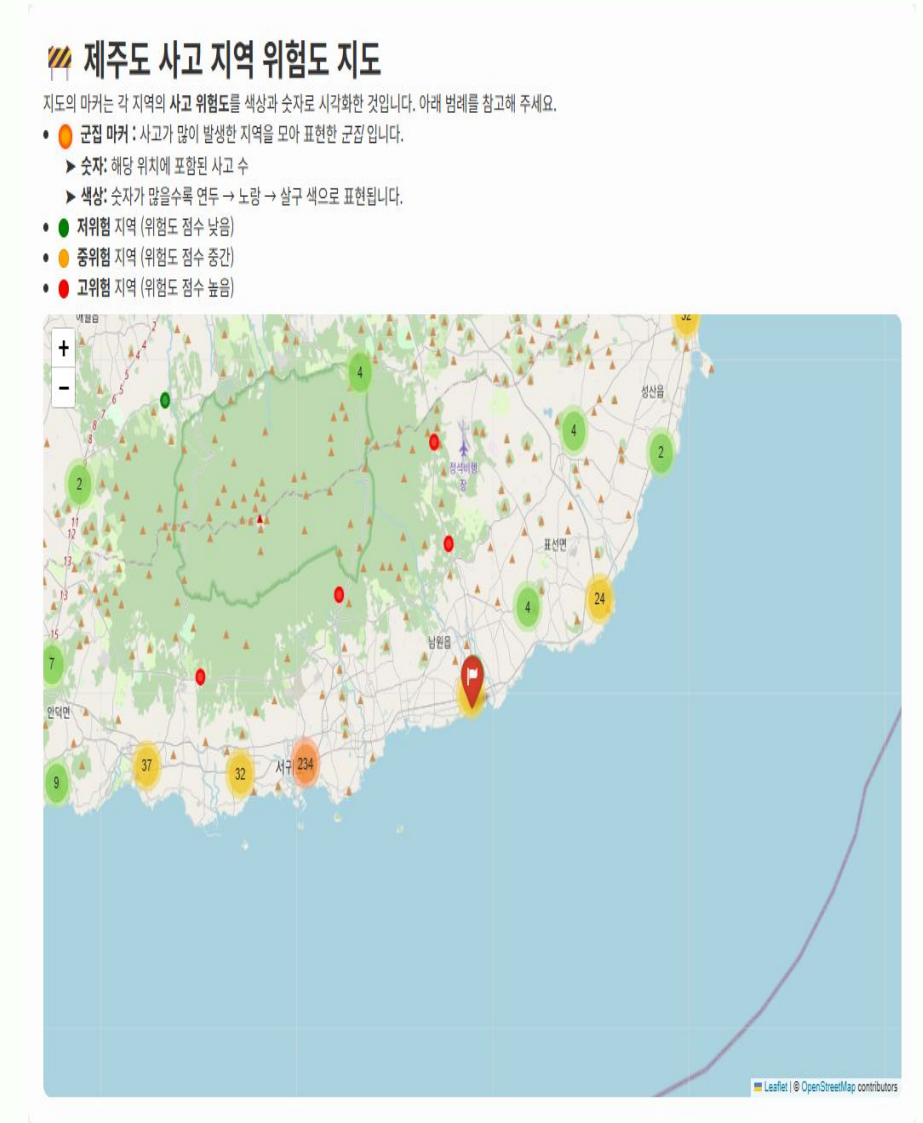
# 주요 구현 화면



대시보  
드



비회원 사고 지역 위험도  
지도



출발지 - 목적지 위험도로 지도



# 사용자 시나리오

## 출발 전 계획

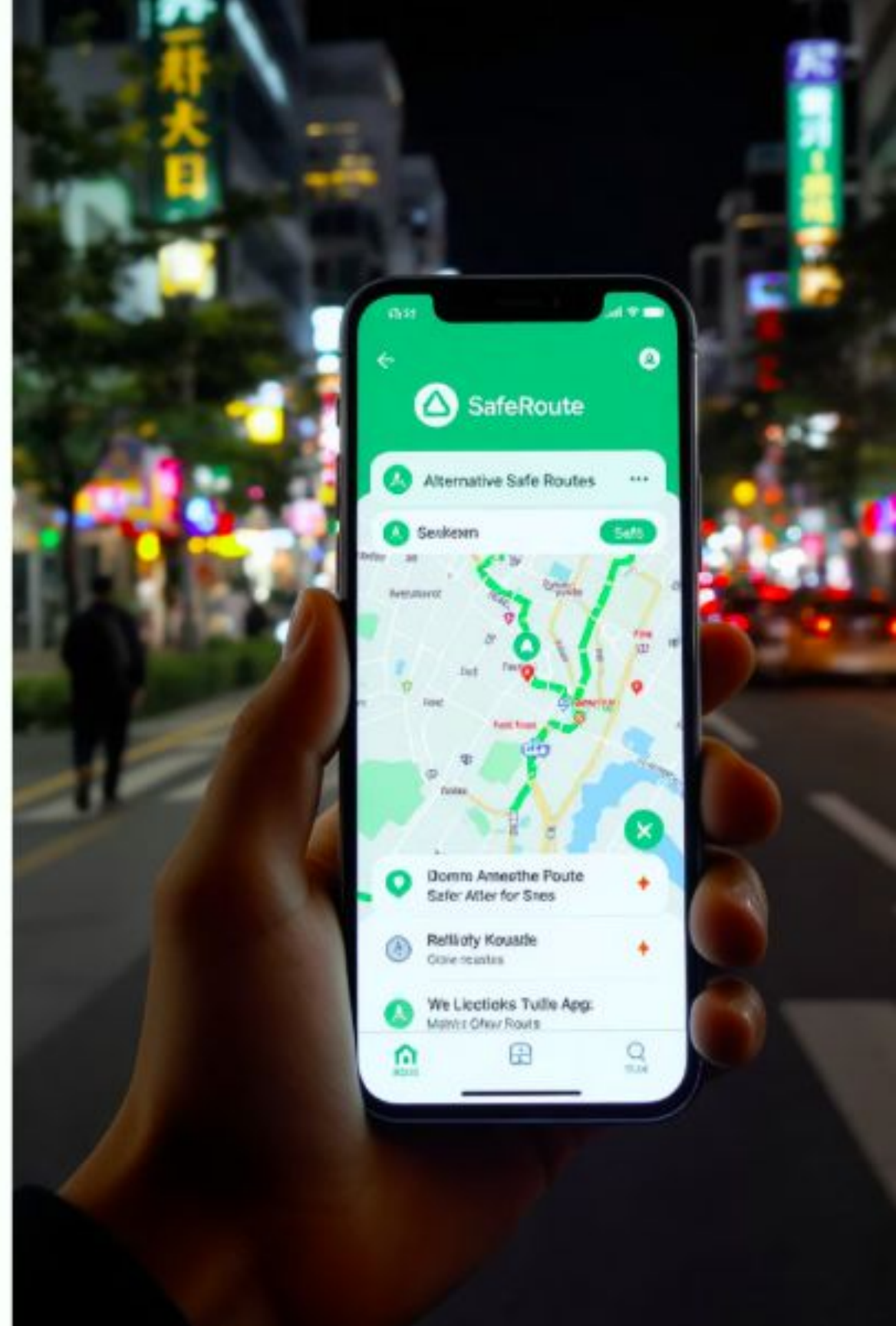
사용자가 웹에서 목적지 입력 시 현재 및 예상 날씨, 시간대를 고려한 최적의 안전 경로 추천

## 이동 중 실시간 안내

경로 이동 중 날씨 변화나 교통 상황 변동 시 실시간 알림 제공 및 대안 경로 제시

## 위험 지역 접근 알림

사고 위험도가 높은 구역 접근 시 특별 경고와 주의사항 안내, 위험 요소 상세 정보 제공





# 기대 효과

## 1 보행자 안전 확보

고위험 구역 실시간 알림을 통해 보행자 사고 예방 및 안전한 보행 경로 추천

## 2 운전자 사고 예방

AI 기반 주의 경로 안내로 위험 구간 사전 인지 및 안전 운전 유도

## 3 데이터 기반 정책 지원

사고 지점 예측 데이터를 활용한 효율적인 교통 안전 정책 수립 및 자원 배분





# 보완 및 개선 계획



## 다양한 시계열 데이터 적용

교통량, 계절 패턴,  
행사 정보 등 추가  
데이터 통합

## 실시간 IoT 센서 연동

도로 상태, 교통량 측정  
센서와 연계하여  
정확도 개선

1

2

## 딥러닝 모델 도입 시도

CNN, LSTM 등 고급  
모델 활용으로 예측  
정확도 향상

3

4

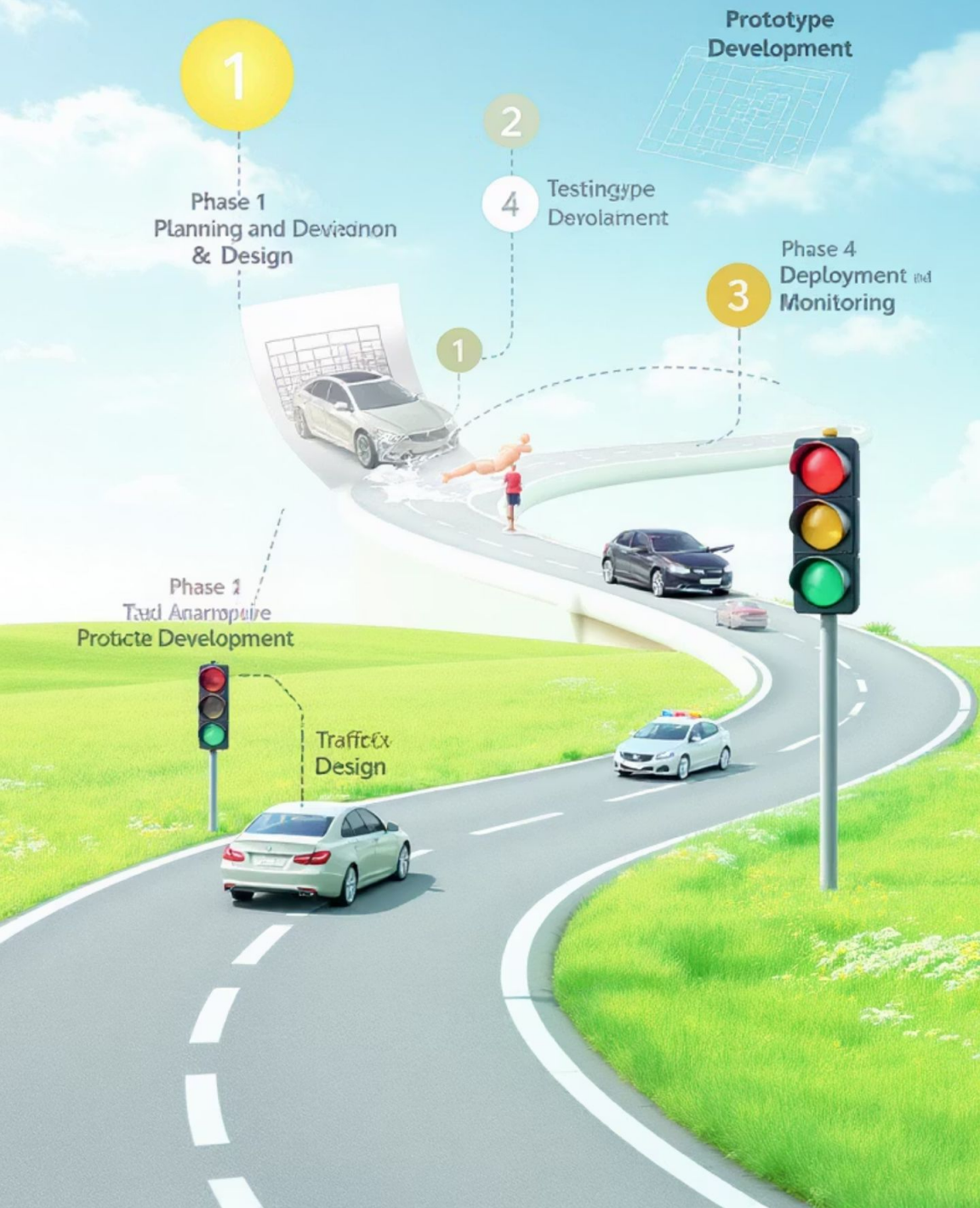
## 위치기반 추천 시스템 확장

개인화된 안전 경로  
추천 및 알림 서비스  
구현



# Project Timeline

for a Traffic Safety System



## 다음 단계 계획

### 🧠 모델 고도화

- 추가 데이터셋 확보
- 모델 성능 개선 및 최적화

### 🌐 지역 확장 및 서비스 운영

- 서비스 지역 확대 및 테스트
- 사용자의 데이터를 수집 및 서비스 개인화
- 사용자의 피드백 반영

### 👤 개인 맞춤형 서비스

- 사용자의 운전 및 보행 데이터 수집
- 사용자의 최근 사용 내역 데이터 수집으로 서비스 개인화

# 감사합니다

## 질문 있으시면

## 말씀해주세요!

스마트 교통 안전 예측 서비스를 통해 더 안전한 도로 환경을 만들어  
나가겠습니다.



# Thank You